

Fibonacci-vierkante kokerklok



Deze vierkante kokerklok is een variatie op de [kast Fibonacci klok](#). De software is vereenvoudigd en geschikt voor twee soorten kleuren-LEDs.

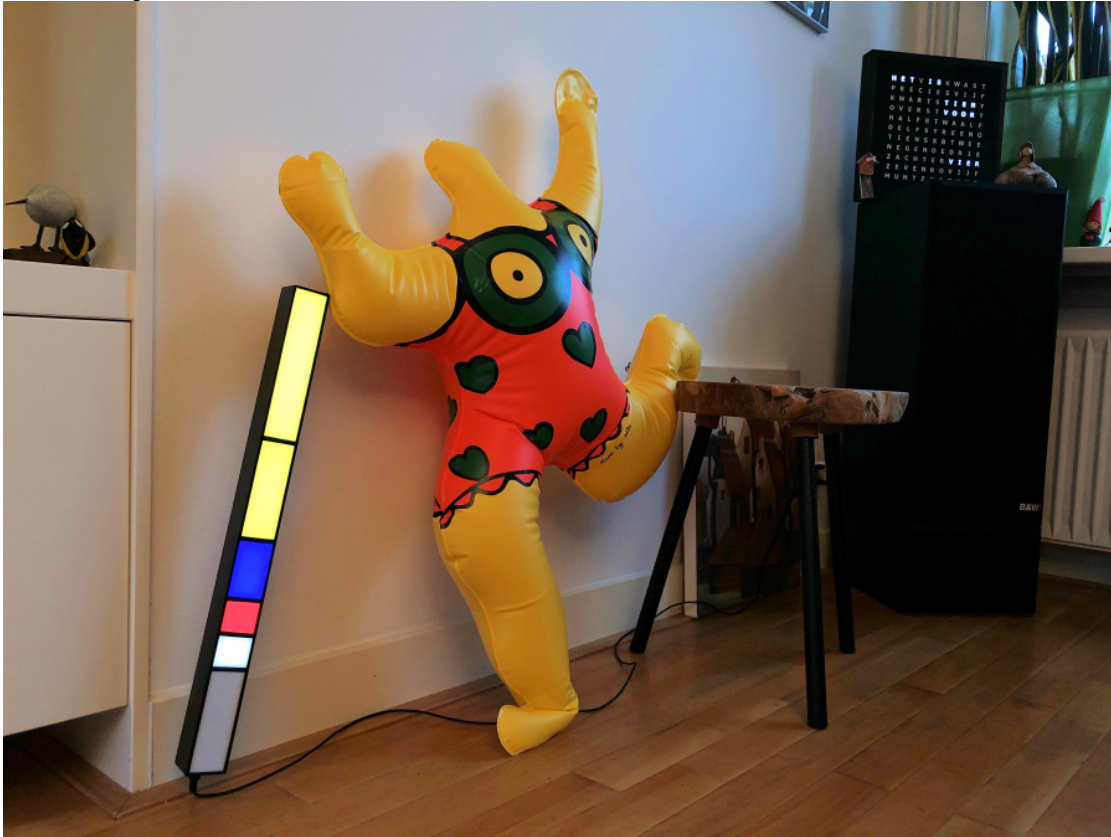
De Fibonacci-klok was een kick-starterproject van [Philippe Chrétien](#).

Wat deze klok zo aantrekkelijk maakt is zijn eenvoud, de vindingrijkheid en de voortdurend veranderende Mondriaan-gelijke schilderijendisplay. Mondriaan was ook op zoek naar eenvoud in zijn schilderijen dat begon bij realisme en via kubisme bij zijn kermerkende stijl eindigde.

Naast het kleurenspeel nodigt deze klok uit tot het narekenen van de gepresenteerde tijd.

Maar een klok? En wat heeft Fibonacci met Mondriaan te maken?

De gekleurde vlakken gescheiden door zwarte lijnen doen meteen denken aan de schilderijen van [Piet Mondriaan](#) als de primaire kleuren rood, geel en blauw worden gebruikt.



In deze klok wordt gebruik gemaakt van kleuren en oppervlakten die de reeks van [Fibonacci](#) volgen.

Rood + Blauw = uur, (Geel + Blauw) x 5 = minuten.

$5 + 3 = 8$ uur, $1 + 3 = 4$, $4 \times 5 = 20$ minuten Het is 8:20, tien voor half negen.

Wat de klok doet is de verlichting aanzetten achter de vlakken die opgeteld moeten gaan worden.

Om de uren en minuten te combineren wordt een derde kleur, blauw, gebruikt.

De rode vlakken zijn de uren.

De gele vlakken vermenigvuldigd met 5 zijn de minuten en de blauwe vlakken als er een rode en gele hetzelfde vlak bezetten.

De witte vlakken betekenen nul, die stellen niets voor met optellen.

Fibonacci was een Italiaanse wiskundige die de naar hem genoemde getallenreeks ontwikkelde. Het was niet een normale getallenreeks maar een die ook vaak in natuurlijke processen voorkomt. Bijvoorbeeld in zonnebloemen en schelpen.

De reeks is simpel. Elk volgend getal in de reeks is de som van de twee voorgaande: 1, 1, 2, 3, 5, 8,

Als wij 1, 1, 2, 3, 5 optellen komen wij op 12.

Ah, precies het aantal uren in een dagdeel en 12 vermenigvuldigt met 5 is het aantal minuten in een uur.

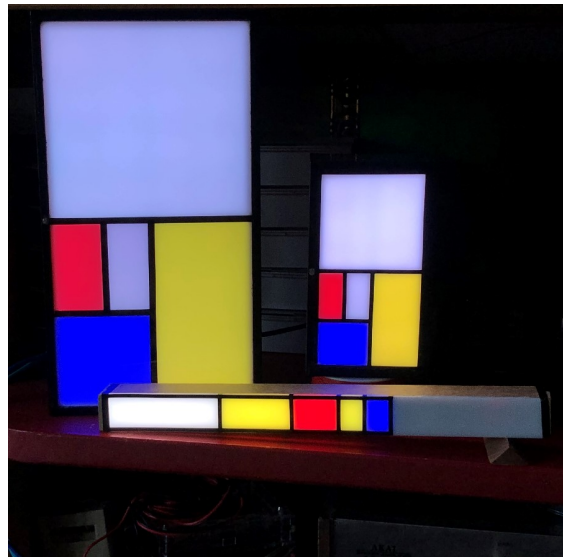
Voorbeeld: 4 uur.

Dat kan een optelsom zijn van : $1+1+2$ of $1+3$.

Voorbeeld: 8 uur.

Dat kan bereikt worden met het optellen van $1+1+2+4$, of $1+3+4$ of $3+5$.

De display van de klok is een weergave van de Fibonacci-reeks en de oppervlakte van de vlakken is de verhouding van de oppervlakte in de reeks.



Als het nu duizelt is het goed.

Nog een voorbeeld. 3:35.

Dat zijn drie rode (uren) vlakken en $35/5 = 7$ gele (minuten) vlakken.

Drie kan je maken met het 1 vlak + 2 vlak of alleen met het 3 vlak.

Zeven kan zijn $1 + 1 + 5$ of $2 + 5$ of $1 + 1 + 2 + 3$.

3:35 tot en met 3:39 geven dezelfde 7 minutenvlakken maar de kloksoftware probeert elke minuut een andere combinatie te vinden. Dit gaat random dus het kan zijn dat hij weer dezelfde combinatie vindt en niet verspringt.

De bouw van de kast

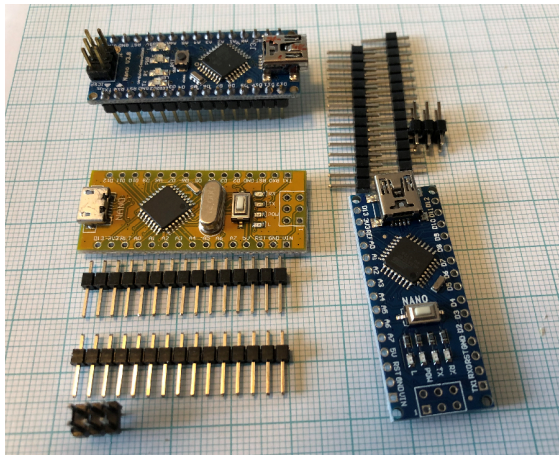
De minimale ruimte die nodig is om de elektronica in een kast te bouwen is $25 \times 25 \times 100$ mm binnenmaten als er een Arduino Nano wordt gebruikt waarvan de aansluitpennen aan een kant zitten.

Maak anders de kast wat hoger en eventueel breder om de zijkant van de kast vierkant te houden

De strip heeft 12 LEDS per 20 cm. De kleinste kast wordt daardoor 30 cm lang.

Bij een kant-en-klare Arduino, zoals hieronder achterin op de foto, zitten de aansluitpennen aan de boven- en onderkant. Dit maakt de kast 15 mm hoger.

Inclusief de aansluitdraden is dan een kast van 40 mm breed of hoog nodig.



De eind 2019 uitgekomen [Nano Every](#) heeft de zes pennen niet meer is daardoor lager (en ook nog goedkoper).

De binnenkant van de compartimenten waar de verlichting in brandt moet perfect wit zijn. Je kan daar bijvoorbeeld MDF voor gebruiken dat wit is. Maar het kan ook eenvoudig door wit papier aan de binnenkant van de compartimenten plakken.

Als scheiding tussen de compartimenten kan dun stevig karton, MDF of ander materiaal gebruikt worden. Maak het niet dikker dan 2 mm.

Het is mooi als de witte perspex bovenplaat in het hout valt.

Zaag hiervoor met een zaagtafel een sleuf van 1 mm diep in de zijkant, 1-2 mm onder de bovenkant ter dikte van de perspexplaat.

Maar het kan natuurlijk ook tussen de zijkanten worden geplakt.

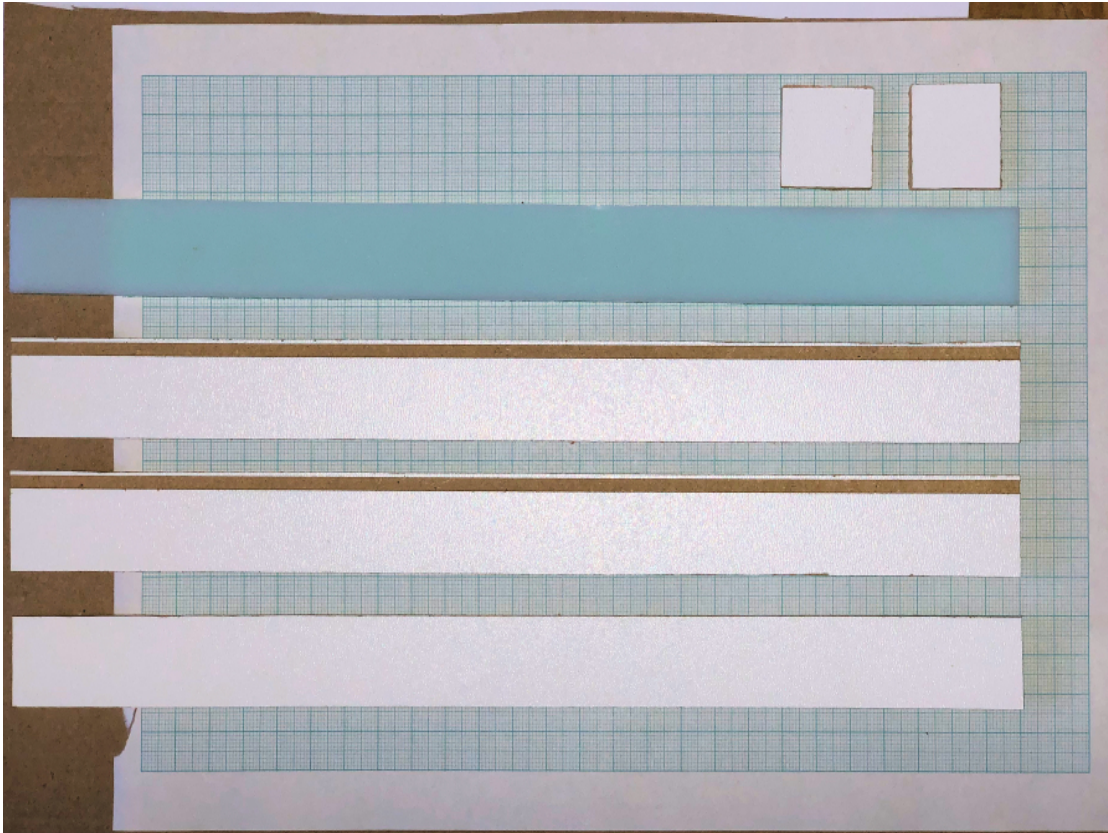
Voor de kleinste kast:

- Zaag een strook voor de bodem 25 mm breed en 300 mm lang.

- Zaag meteen een strook van 25 mm breed (maximaal 2 mm dik) voor de afscheidingen voor de compartimenten en zaag de zeven afscheidingen op maat (25x25mm).

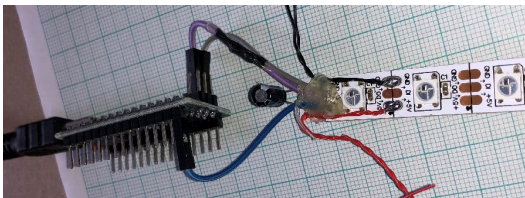
Als de compartimentafscheidings uit karton worden geknipt kan dat later.

- Zaag twee zijanten van 35 cm op een breedte van 25 mm + de dikte van de bodemplaat.
 - Zaag eventueel de sleuf uit waarin het perspex moet gaan vallen.
 - Zaag de zijanten van de kopse kant uit met een breedte van 25 mm.
 - Plak wit papier op alle tussenschotten en zijanten van de kast waar de verlichting komt.
- &
- Plak de kast in elkaar op een kopse kant na. Handig! Gebruik papieren afplakband om de onderdelen stevig op elkaar geklemd te houden.
 - Plak de LED-strip, van de gelijmde kopse kant af, op de bodem vast.
 - Breng aan de zijanten van de strip een druppel Bison-kit aan om te voorkomen dat de zijkant van de strip in de loop der jaren opkrult.
 - Reng de elektronica aan (zie hieronder) en lijm de tussenschotten vast.
 - Werk de kast netje af als de elektronica naar behoren werkt
 - Plak met 2 mm breed zwart plakband of teken met een zwarte viltstift de compartimentscheidingen af op het perspex voor een beter contrast tussen de lichtvlakten



(Zie foto bovenin deze pagina)

Voordeel van de grotere kast is tevens dat er dan een grotere en nauwkeuriger DS3231-tijdsmodule gebruikt kan worden.



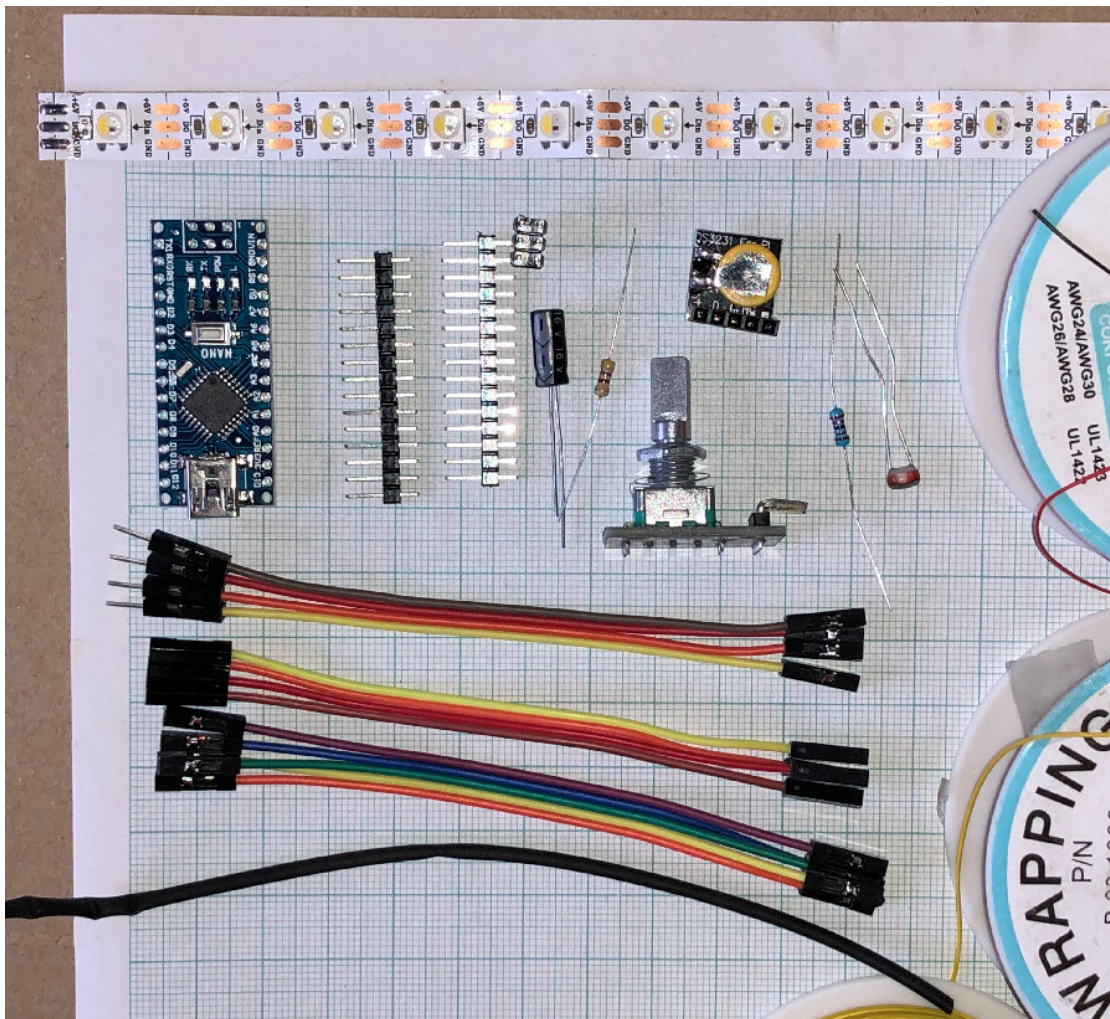
De strong>De elektronica in elkaar zetten

De benodigdheden voor de elektronica is hieronder weergegeven

Een soldeerbout is nodig om drie draden aan de LED-strip te solderen.

De lichtgevoelige sensor en de weerstand zouden bij voorkeur ook gesoldeerd moeten worden.

Een multimeter van 10 euro is handig om de weerstanden te kunnen meten en eventuele slechte verbindingen te controleren.



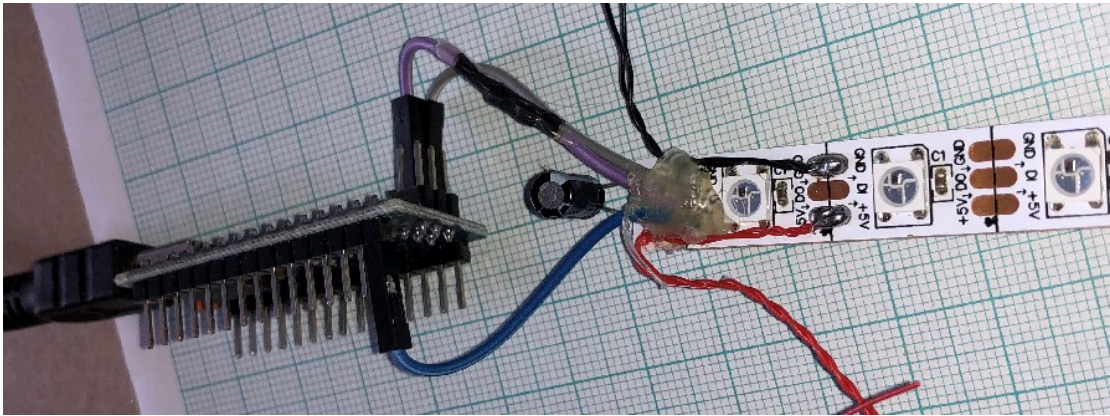
Benodigd is:

- WS2812 of SK6812 LED-strip met twaalf LEDs
 - [Arduino Nano](#) of [Arduino Nano Every](#)
 - DS3231 RTC-module.(1) DS3231 for Pi of DS3231 RTC
 - 100 - 1000 uF condensator
 - 330 of 470 ohm weerstand
 - 22 Kohm weerstand
 - LDR (lichtgevoelige weerstand)
 - Rotary encoder
 - 9 Dupont-kabeltjes vrouw-vrouw
 - 4 Dupont-kabeltjes man-vrouw
 - Isolatieband of krimpkous met diameter van ongeveer 2 mm
 - Een paar stukjes draad
 - Soldeerbout en soldeer
- (1) In de kleinste klok past alleen de "DS3231 for Pi" klokmodule.

Hieronder is het aansluitschema getekend.

Bij [Fritzing](#) kan het programma gedownload worden om deze schema's te ontwerpen

Click op het plaatje voor het ontwerpbestand dat in het Fritzing-programma kan worden ingelezen en bewerkt.



Het is verstandig om de draden te labelen. Dit voorkomt fouten bij het aansluiten en is praktisch mocht er ooit iets later losraken of vervangen moeten worden.

- Label alle draden dus zoals aangegeven in het Fritzing-ontwerpschema hierboven.

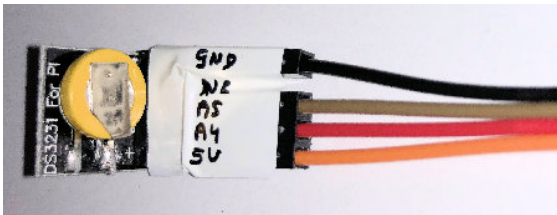
Het klopt dat de DS3231-klokmodule op 3.3V wordt aangesloten maar kan ook op 5V worden aangesloten.

De rotary ontvangt zijn voeding van een digitale pin die in de software "aan" wordt gezet.

`pinMode(EncoderPow, OUTPUT);`

`digitalWrite(EncoderPow, HIGH);` Er staat dan 5V op pin 2 en daar kan veilig 20 mA stroom van worden getrokken

De klok zou eventueel ook met een digitale pin kunnen worden gevoed.

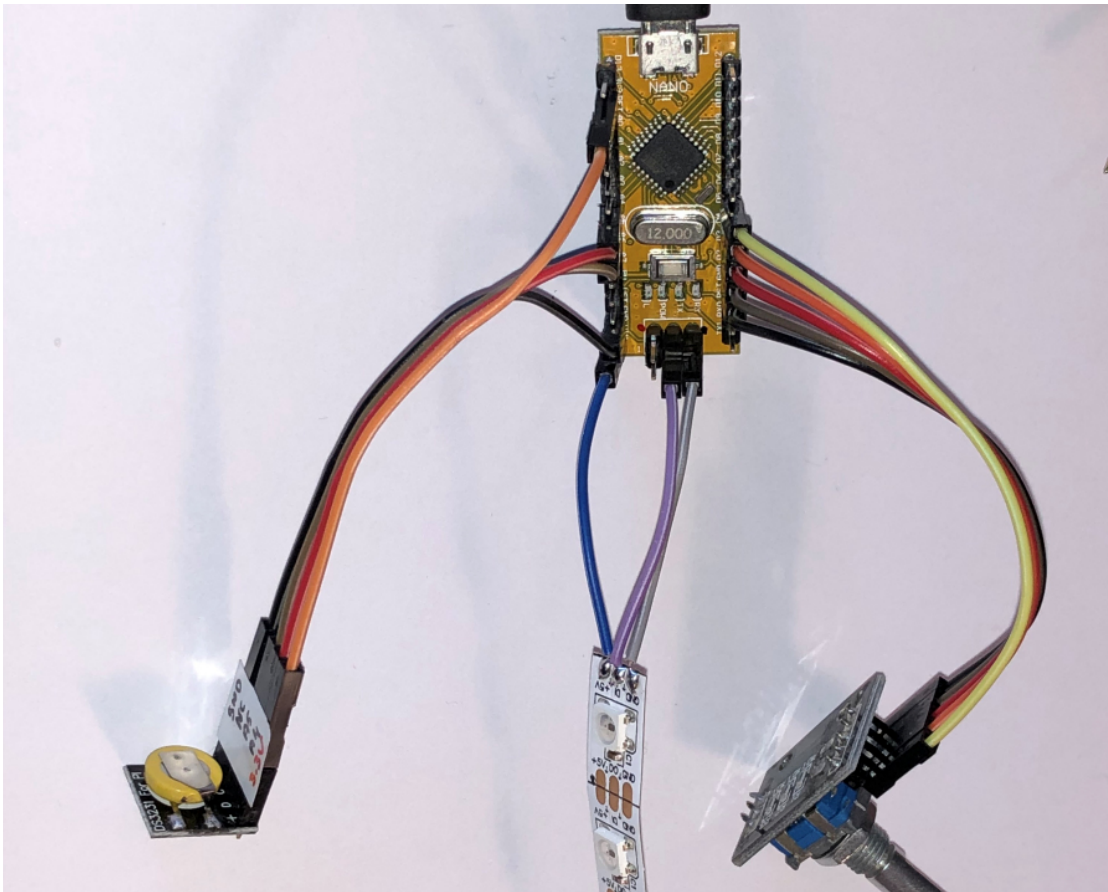


- Sluit de LED-strip en eventueel de LDR aan zodat de elektronica getest kan worden.

Op de LED-strip tussen GND en 5V is de weerstand meer dan 1 Mohm (1 miljoen ohm).

Tussen 5V en GND pennen op de Arduino Nano is het meer dan 10 kOhm (10,000 Ohm).

- Sluit de Rotary en RTC-klokmodule aan volgens schema hierboven



- Sluit de USB-stekker aan op een 5V-voeding.
Wacht een paar seconden. Geen brandlucht en de LEDs gaan branden?

De software

Vorbereiding IDE

- Download de Arduino IDE van: <https://www.arduino.cc/en/Main/Software>

ARDUINO 1.8.10

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software. This software can be used with any Arduino board. Refer to the Getting Started page for installation instructions.

Windows installer for Windows XP and up
Windows ZIP file for non-admin install

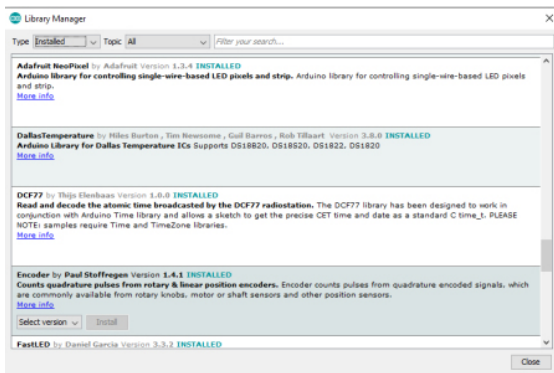
Windows app Responses 8.1 or 10
Get it

Mac OS X 10.8 Mountain Lion or newer

Linux 32 bits
Linux 64 bits
Linux ARM 32 bits
Linux ARM 64 bits

Release Notes
Source Code
Checksums (SHA256)

De software maakt gebruik van libraries. Dit is software door anderen geschreven welke functies bevat waar je anders maanden mee bezig was om uit te zoeken en te programmeren. Nu kan je dat werk dat anderen voor je uitgezocht hebben gebruiken.



De Arduino-programmeeromgeving biedt een groot scala aan libraries aan.

Om deze Fibonacci-kloksoftware te kunnen aanpassen moet in de Arduino IDE de volgende libraries geïnstalleerd worden:

dit gaat als volgt:

- In het IDE-menu open: Sketch->Include library -> manage libraries

- Zoek de volgende libraries en installeer ze

Adafruit Neopixel

Encoder by Paul Stoffregen

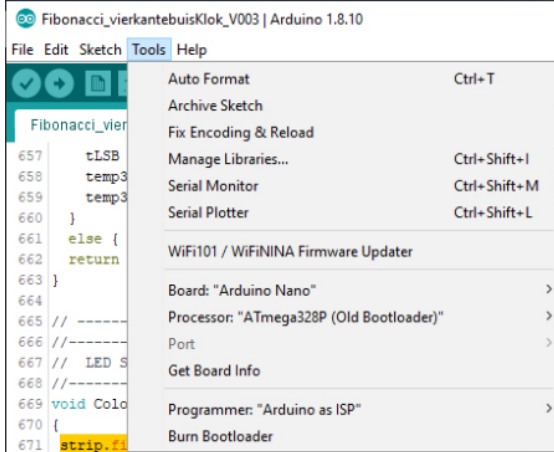
Keypad by Mark Stanley, A Brevig

RTCLib bij Adafruit

Er zijn veel soorten Arduino's. wij gebruiken de Arduino Nano

In het menu: Tools -> board kan je de Arduino Nano selecteren

Kies bij Processor ATMEGA328p (Old bootloader)



Om de code in de Arduino Nano te plaatsen moet een Arduino via een de USB-kabel aan de PC worden aangesloten.

Port wordt dan zwart en er kan een compoort worden aangevinkt

Als daarna de pijl naar rechts onder edit in het menu wordt gedrukt wordt het programma in de Arduino nano geladen.

De coding

Met de definities kunnen de gebruikte onderdelen geselecteerd worden.

Door twee slashes // voor een #define te zetten wordt de optie uitgeschakeld.

In de coding kan je herkennen aan de #ifdef en #endif. Het stukje code wordt niet gecompileerd in de software.

```
#ifndef ROTARYMOD
#include <Encoder.h> // For rotary encoder
#endif ROTARYMOD
```

De software ondersteunt WS2812 of SK6812 LED-strips. Kies een van deze.

Daarnaast kan er gekozen worden tussen een draaiknop (ROTARYMOD) of een keypad (KEYPAD).

De klok kan zonder tijdmodule draaien. Het werkt dan met de klok van de processor.

Deze processorklok heeft een afwijking van seconden per dag. Dit is alleen handig om de software te testen.

Een DS3231-module heeft een afwijking van slechts een minuut per jaar.

Als #define MOD_DS3231 is uitgezet wordt de processorklok gebruikt.

```
//-----
// ARDUINO Definition of installed modules
//-----
//#define LED2812 // Use RGB LED strip WS2812
#define LED6812 // Use RGBW LED strip SK6812
#define ROTARYMOD // Use rotary encoder
```



```

// #define KEYPAD           // Use a 3x4 keypad
#define MOD_DS3231         // DS3231 RTC module installed

// -----
// ARDUINO Includes defines and initialisations
// -----
#ifdef ROTARYMOD
#include <Encoder.h>        // For rotary encoder
#endif
#include <Wire.h>           // Communication with
#include <RTClib.h>         // For RTC module
#include <EEPROM.h>         // To store data in EEPROM
#include <TimeLib.h>        // For time management
#include <Adafruit_NeoPixel.h> // for LED strip WS2812 or SK6812

#ifdef KEYPAD
#include <Keypad.h>        // For 3x4 keypad
#endif

```

Bij "Pin Assignments" wordt een leesbare naam voor een pin gedefinieerd.

Digitale pinnen kunnen alleen aan of uit worden gezet of uitgelezen

Bij analoge pinnen kunnen voltages tussen 0 en 5 V worden gemeten of gestuurd met waarden tussen 0 en 1024

```

// -----
// PIN Assignments
// -----

enum DigitalPinAssignments {
  EncoderPow   = 2,      // give power to Encoder
  clearButton  = 3,      // switch (labeled SW on decoder)
  encoderPinA  = 4,      // right (labeled DT on decoder)
  encoderPinB  = 5,      // left (labeled CLK on decoder)
  EmptyD06     = 6,      // EmptyD06
  EmptyD07     = 7,      // EmptyD07
  EmptyD08     = 8,      // EmptyD08
  EmptyD09     = 9,      // EmptyD09
  EmptyD10     = 10,     // EmptyD10
  LED_PIN      = 11,     // Pin to control colour SK6812 WS2812 LEDs MOSI
  EmptyD12     = 12,     // EmptyD12
  secondsPin   = 10;     // if set to 13 led will blink on board SCK
                  // Analogue hardware constants ----

enum AnaloguePinAssignments {
  PhotoCellPin = 2,      // LDR pin
  EmptyA3      = 3,      // EmptyA3
  SDA_pin      = 4,      // SDA pin
  SCL_pin      = 5;      // SCL pin
}

```

Hierna volgen nog meerdere definities van variabelen

De Arduino start na de initialisaties de functie setup() na aanzetten.

Hierin wordt van alles aangezet.

Daarna gaat wordt de functie loop() eindeloos doorlopen. Dit is het hart van de coding .

```

// -----
// ARDUINO Loop
// -----
void loop(void)
{
  SerialCheck();
  if(Demo) Demomode();
  else
  {
    EverySecondCheck();
    if(!KeyInputactivated) EveryMinuteUpdate(); // if keyboard input then do not update display
    RotaryEncoderCheck();
  }
}

// -----
// ARDUINO Setup initialise the hardware
// -----
void setup() // initialise the hardware // initialize the appropriate pins as outputs:
{
  Serial.begin(9600); // setup the serial port to 9600 baud

  pinMode(secondsPin, OUTPUT);
  #ifdef ROTARYMOD
  pinMode(encoderPinA, INPUT_PULLUP);
  pinMode(encoderPinB, INPUT_PULLUP);
  pinMode(clearButton, INPUT_PULLUP);
  pinMode(EncoderPow, OUTPUT);
  digitalWrite(EncoderPow, HIGH); // Provide the rotary encoder with power
  Tekstprintln("Rotary encoder enabled");
  #endif

  strip.begin(); // Start communication to LED strip
  strip.setBrightness(BRIGHTNESS); // Set brightness of LEDs
  ShowLeds();
}

```

et cetera

De software kijkt razendsnel steeds of er wat serieel wordt ingetikt, of de demo stand is aangezet en of er aan de draaiknop wordt gezeten. Verder zijn er nog eens per seconde en eens per minuut acties die de overige aandacht van de software eist.

De software is voorzien van commentaar en wijst zichzelf.
Succes

Ed

Bijlagen

[Arduinostoksoftware V002](#) als ZIP-file

[Software op Github](#)

10 dec 2019

Ed Nieuwenhuys